JOHNSON
GRANT
IN-81-CR
252531
13P.

(NASA-CR-186006) DEVELOPING INTEGRATED PARAMETRIC PLANNING MODELS FOR BUDGETING AND MANAGING COMPLEX PROJECTS (Houston Univ.) 13 p CSCL 05A

N90-14986

Unclas
G3/81 0252531

Developing Integrated Parametric Planning Models for Budgeting and Managing Complex Projects

Vance A. Etnyre
Ken U. Black

University of Houston - Clear Lake

April 18, 1988

Cooperative Agreement NCC 9-16
Research Activity No. ***IM.7***

Research Institute for Computing and Information Systems
University of Houston - Clear Lake

T·E·C·H·N·I·C·A·L R·E·P·O·R·T

The RICIS Concept

The University of Houston-Clear Lake established the Research Institute for Computing and Information systems in 1986 to encourage NASA Johnson Space Center and local industry to actively support research in the computing and information sciences. As part of this endeavor, UH-Clear Lake proposed a partnership with JSC to jointly define and manage an integrated program of research in advanced data processing technology needed for JSC's main missions, including administrative, engineering and science responsibilities. JSC agreed and entered into a three-year cooperative agreement with UH-Clear Lake beginning in May, 1986, to jointly plan and execute such research through RICIS. Additionally, under Cooperative Agreement NCC 9-16, computing and educational facilities are shared by the two institutions to conduct the research.

The mission of RICIS is to conduct, coordinate and disseminate research on computing and information systems among researchers, sponsors and users from UH-Clear Lake, NASA/JSC, and other research organizations. Within UH-Clear Lake, the mission is being implemented through interdisciplinary involvement of faculty and students from each of the four schools: Business, Education, Human Sciences and Humanities, and Natural and Applied Sciences.

Other research organizations are involved via the "gateway" concept. UH-Clear Lake establishes relationships with other universities and research organizations, having common research interests, to provide additional sources of expertise to conduct needed research.

A major role of RICIS is to find the best match of sponsors, researchers and research objectives to advance knowledge in the computing and information sciences. Working jointly with NASA/JSC, RICIS advises on research needs, recommends principals for conducting the research, provides technical and administrative support to coordinate the research, and integrates technical results into the cooperative goals of UH-Clear Lake and NASA/JSC.

Developing Integrated Parametric Planning Models for Budgeting and Managing Complex Projects

Preface

This research was conducted under the auspices of the Research Institute for Computing and Information Systems by Vance A. Etnyre, Associate Professor of Information Systems and Quantitative Methods, and Ken U. Black, Interim Program Coordinator and Associate Professor of Information Systems and Quantitative Methods, of the University of Houston - Clear Lake.

Funding has been provided by the Space Station Project Control Office, NASA/JSC through Cooperative Agreement NCC 9-16 between NASA Johnson Space Center and the University of Houston - Clear Lake. The NASA Technical Monitor for this activity was H. Wayne Whittington, Space Station Project Control Office, NASA/JSC.

RICIS Grant IM 7

Developing Integrated Parametric Planning Models
for Budgeting and Managing Complex Projects

Project Status Report

April 18, 1988

Dr. Vance A. Etnyre

and

Dr. Ken U. Black

ORIGINAL PAGE IS
OF POOR QUALITY

RICIS Grant IM 7

Developing Integrated Parametric Planning Models for Budgeting and Managing Complex Projects

Purpose of Study

The primary purpose of this study will be to demonstrate the applicability of integrated parametric models for the budgeting, and management of complex projects. The characteristics of integrated parametric models will be investigated and compared with other methods often used in planning and managing complex projects. A prototype system will be built to allow project planners to design and test various project planning strategies interactively.

Specific integration methodologies using parametric analysis and interactive decision rules will be developed. Specifications will be stated for the development of software systems which can integrate the functions of forecasting, budgeting, and resource management.

Background of Study

This study was originally proposed in May of 1987 as a mechanism to develop and test project costing models and project management strategies for the U. S. space station. Cost estimators and managers had expressed interest in having very flexible costing models which could adjust to abrupt changes in project funding levels, project duration requirements, and project budget requirements. In the past, several mechanisms have been envisioned for developing project cost estimates. In May of 1987, however, much of the costing work was being done manually due to the rapidly changing budgetary environment and the uncertain nature of the funding process.

Project Goals (General)

This study has been subdivided into two phases. The primary goal of the first phase was to build a very flexible, interactive prototype for a project planning system. This prototype should provide most of the important features of a project scheduling system in a flexible, interactive manner. The prototype must allow the user to build and test a wide variety of parametrically defined cost models.

The primary goal of the second phase is to use the prototype developed in Phase 1 as a design tool to develop and test sophisticated project management strategies. These strategies must be able to handle flexible project schedules, variable project costs, and meet dynamically changing constraints. The system must be able to capture the essential cost characteristics of existing or proposed projects. It also must allow project management experts to interactively define and test procedures to be used in the management of complex projects.

Specific Requirements for Phase 1

The general requirements for the prototype were generated from discussions between the project sponsor and the project investigators. A very basic description of the requirements for Phase 1 is contained in the following list:

1. The system must accept the information necessary to define the project in a user-friendly manner.

2. The system must be capable of calculating schedule and loading information (Gantt Charts & Loading Diagrams) once the necessary task information has been input by the user.

3. The system must be capable of displaying project structure, loading information, tasks and sub-tasks in a dynamic, interactive manner.

4. The system must be capable of printing and plotting the schedule information (task inputs, Gantt charts, & Loading diagrams).

5. The system must be capable of storing and retrieving various portions of the project structure and combining these portions to form larger pieces of the project cost structure.

Specific Requirements for Phase 2

1. The system must allow the user to modify the essential details of any task or subtask within the project in an easy, user-friendly manner.

2. The system must allow the user to modify the structure of the project in an easy, user-friendly manner.

3. The system must be capable of printing and plotting revised schedule information (task inputs, goals, constraints, Gantt charts, and loading diagrams) in an easy, user-friendly manner.

4. The system must allow the user to define the cost characteristics of any project in a parametric fashion using very general (arbitrary) cost functions.

5. The system must be capable of capturing the essential cost characteristics of any project, existing or proposed directly from the cost data.

6. The system must allow project management experts to define interactively the procedures needed to manage a project.

Evaluation of Methodologies

There are several cost estimation methodologies available to project managers. A partial listing would include: group methods, detailed methods, comparison methods, order-of-magnitude methods, approximate estimates, definitive estimates, industrial engineering methods, analogy methods and statistical methods.

Detailed methods require that the job be completely specified with detailed estimates prepaired for each item, each subassembly, and for the overall project. Industrial engineering approaches are based on highly detailed applications of time standards. They involve thorough analyses of each task so that workhours can be converted to project cost estimates. These methods are very expensive and, may not be feasible for technologically innovative efforts.

Comparison methods utilize up-to-date estimates of cost based on projects already in existence. For these methods to be effective, the estimator must be able to extract the essential cost characteristics from existing projects. Approximate estimating and analogy methods use top-down estimates prorated from similar projects. Little detailed engineering data is required as the analogies are made using rules of thumb and indexed costs from similar projects. Although these methods can be very useful when similar projects are available, they are not very useful for purely innovative projects.

Statistical cost estimating methodologies involve parametric cost estimation procedures which require very restrictive assumptions about the nature of cost relationships within the project. These methods, which use computerized data analysis techniques, can provide valuable information in areas where sufficient examples of similar projects exist. Like comparison and analogy methods, however, they are of limited use for purely innovative projects.

Many of these techniques are not viable for use in estimating costs of large, innovative projects such as those proposed for the space station. It is often impossible to acquire the amount of engineering detail necessary to use various detailed methods when significant proportions of the project are still in the conceptualization and design phases. It is difficult to apply statistical or analogy methods to projects which have never before been created.

Complex and innovative developmental projects present unique problems in forecasting, budgeting, and resource management. Reliable models for forecasting resource requirements usually do not exist for developmental projects. Inherent dissimilarities between various developmental projects limit the usefulness of analogy methods and generalized budget and planning packages. In order to handle innovative developmental projects, a framework must be created which uses the positive aspects of the methodologies discussed above, but avoids the known pitfalls of those methods.

Evaluation of Software Resources

Several types of software were considered for constructing the prototype system and testing project management strategies. Among the categories considered were: stand-alone (dedicated) project management systems; procedural programming language systems; database management systems; declarative language systems; spreadsheet programs and special programming systems.

Stand-alone (dedicated) project management systems were considered because they already contain many of the elements needed for effective project management. Systems such as Artemis Project Management system, Primivera Project Management System and Harvard Total Project Management System allow the user to enter data to describe the various activities within a project. These systems then use the defined structure of the project and critical path methodology to determine the performance and cost characteristics of the project. Certain of these software packages allow the user to perform "load leveling" to adjust certain cost characteristics of the project to accomodate required changes in task durations. The major disadvantages of stand-alone project management packages is the lack of flexibility that they present to the user. Only certain cost loading patterns are allowed in these models. Although most of these packages allow the user to enter actual project performance data for comparison to the estimated values generated by the package, none of the packages examined allow for the automatic capturing of project cost characteristics from actual data with translation directly into a fully parametric model. Although each package allows the user to interface with one or more procedural programming languages, none allows the user to define procedures in a syntax which the user can design for himself.

Several procedural programming languages were considered, but all were considered inappropriate for the development of this protoype. Ultimately, the users of this system will be project managers and not programmers. Although many parts of the system can be preprogrammed for the user, there will be several key areas where the project planning experts will have to define the procedures to be used in an interactive manner. This eliminates the use of current procedural languages for developing and interactively testing project management strategies.

Many database systems have included programming language interfaces in so-called "fourth generation" languages (4GLs) to aid in the prototyping of systems. These systems can be extremely useful when the primary difficulties in system design are caused by complex data flows. For project planning systems which are still in the conceptualization and design phases, it is the lack of valid data sources rather than the complexity of existing data sources which is the primary limitation on the system.

Declarative languages such as Prolog and LISP provide useful features for developing relationships between various sets of objects. Although these languages do provide facilities for mathematical calculations and graphical displays, the specialized skills needed to utilize these features effectively are beyond the experiences of most project managers.

Special program development environments were also considered. Most of these special program development environments were rejected due to lack of user familiarity with the development systems. The package IFPS (Interactive Financial Planning System) was particularly noteworthy in the features which it offered for developing prototypes of this nature.

Spreadsheet programs have the advantage that most project managers are familiar with the basic commands and features. This greatly simplifies the task of providing user interfaces. The inherent two-dimensional character of spreadsheet programs and their built-in graphic interfaces allow great flexibility in displaying data. Some spreadsheet packages include a complete set of programming commands and development tools to simplify the process of creating customized operating environments. If the set of programming tools is sufficiently large, a knowledgeable designer can use these tools to create any model which could be created in any other programming environment.

A fundamental disadvantage of spreadsheet packages is the inherently slow execution of interpretive language when compared to the execution of compiled instructions. Historically, spreadsheet systems have depended on interpretive execution of "macro-instructions" supplied by the designer. This has caused spreadsheet programs to be limited to environments where execution time is not a significant design factor.

One of the significant developments in spreadsheet program features is the development of spreadsheet compilers. These allow the designer to compile into efficient code the parts of his logic which are critical in execution time. The remaining parts can retain all the flexibilty allowed by the spreadsheet system.

Prototype Development Strategy

The decision to develop the prototype system using a spreadsheet package (Lotus 123 - release 2) was based on two primary criteria: familiarity and flexibility. The potential users of the prototype are already familiar with the basic mechanics of Lotus 123. This means that two of the most important aspects of the user interface, data entry and selection of options, will be immediately useable with no need for specialized training. Several of the potential users were already familiar with Lotus 123's internal programming language. Ultimately, the decision as to whether to transport the models into another programming system will depend on the execution time requirements and the availability of efficient spreadsheet compilers.

Proposed Solution

The development of a project planning prototype has been formulated in terms of an integrated, open-ended parametric programming problem.

The requirement for an integrated system was included because several aspects of the project development process must be considered. These aspects include starting and stopping times for the project and for the individual tasks that make up the project. The starting and stopping times must be related to other considerations such as total project cost, cumulative cost through various portions of the project, and maximum expenditure per time period.

The open-ended requirement is necessary because the exact nature of the project's goals will vary during the development of the project. The requirement for a parametrically driven prototype is necessary because of the rapidly changing environment in which innovative developmental projects must operate. The prototype must be sensative to changing objectives, changing target dates, changing cost relationships and changing budget constraints.

In order to achieve the integration of costs and project and task durations, parametric cost functions must be defined. In general, such functions can be defined by a process of trapezoidal segmentation. In such a system, the total cost for the project is the sum of the various project cost segments, and each project cost segment is the integral of a linearly segmented cost loading function over a specific interval. Algebraically, this can be stated as:

$$\text{project cost} = C_{project} = \sum_{i=1}^{n} C_i$$

$$\text{cost per segment} = C_i = \frac{f_{i-1} + f_i}{2} * w_i$$

$$= [\, f_o + \sum_{j=1}^{i-1} r_j w_j \,] * w_i + 1/2\ r_i w_i^2$$

$$\text{cost per time period} = f_i = f_{i-1} + r_i w_i = f_o + \sum_{j=1}^{i} r_j w_j$$

Within this framework, important constraints can be defined by parametrically assigning values to key costs and durations. Then specific goals can be achieved by finding the proper combination of the unconstrained variables w_i and r_i which satisfy the defined objectives within the defined constraints.

Prototype Design

The prototype was designed using Lotus-123 (Release 2) as the primary software tool. It was implemented using Release 2 macro instructions and system menus customized for this application. The menu system was used to provide the user interface to the program. A combination of Lotus-123 formulas and macros was used to implement the other requirements of the system. Flexibility and ease of use were the primary design determinants. An additional design requirement was to allow an indefinite number of potential tasks and sub-tasks. Another important design consideration was to place no restrictions on the number of task levels used in the system.

This project was designed in two sequential phases. In the first phase, a simplified prototype was proposed for the integrated model. Goals were specified and solutions were formulated for this simple, integrated prototype. The schedule proposed for Phase One of this study is shown below in figure 1.

Schedule for Phase 1

```
Projected
Values                Jun-87   Jul-87   Aug-87   Sep-87   Oct-87   Nov-87   Dec-87   Jan-88   Feb-86

PHASE 1
=======
Requirements         !!!!!!!!!!!!!!!!!!
Design strategy               !!!!!!!!!
Input requirements            !!!!!!!!!!!!!!!!!!!!!!!!!!!
Loading process                        !!!!!!!!!!!!!!!!!!!!!!!!!!!
Charts & printing                                                 !!!!!!!!!!!!!!!!!!
Integrate: menus                                                           !!!!!!!!!!!!!!!!!!
Demonstrate                                                                         !!!!!!!!!
Revise                                                                              !!!!!!!!!!!!!!!!!!
Wrap-up                                                                                      !!!!!!!!!
                      ------   ------   ------   ------   ------   ------   ------   ------   ------
Phase 1.             !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

figure 1

Summary of Progress (Phase 1)

A prototype system has been implemented for phase 1 of this project using Lotus-123 (Release 2) macro language. This prototype implements a methodology for interactive project scheduling. The prototype provides a model of a system which is capable of meeting most of the goals for Phase 1 of this study, and several of the goals specified for Phase 2.

ORIGINAL PAGE IS
OF POOR QUALITY

Schedule for Phase 2

In the second phase, specific project adjustment procedures will be formulated and integrated into the model in a parametric manner. This will allow NASA managers to change project durations, peak loading requirements or specific performance requirements for any part of the project, and receive "nearly immediate" feedback about the important characteristics of the project being planned. Decision rules will be developed for making necessary adjustments to the project. Methodologies will be proposed for extending the solution of the simple, integrated prototype to include the user-defined adjustments to the project.

Phase 2 of this project, which would begin June 1, 1988 will take five months to complete. A version of the integrated model, containing features from both Phase 1 and Phase 2, would be available by October 31, 1988.

PHASE 2

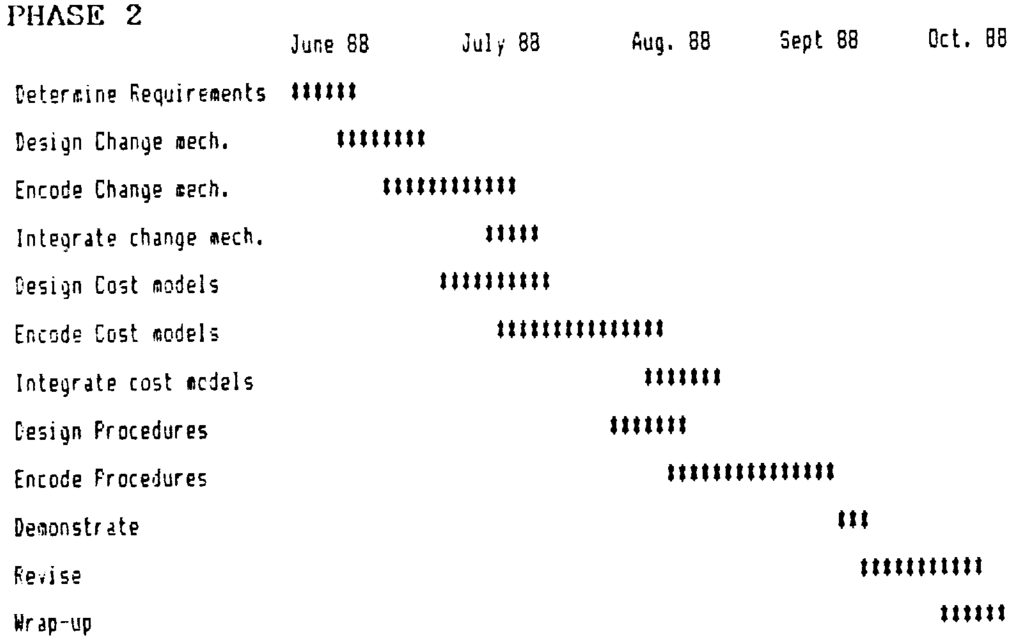

figure 2

Budget for Phase 2

When this project was first proposed in May of 1987, a two-year combined budget was submitted. The attached budget for Phase 2 during 1988 is a minor modification of the Phase 2 budget submitted in 1987. The slight modification is necessary because Dr. Ken Black will not receive summer pay for his work on the second phase of this project. Dr. Vance Etnyre will receive three months of summer pay for his work on the second phase of this project.